Evaluating Sampling Strategies for Similarity-Based Short Answer Scoring: a Case Study in Thailand

Pachara Boonsarngsuk*, **Pacharapon Arpanantikul***,
Supakorn Hiranwipas, **Wipu Watcharakajorn**, **Ekapol Chuangsuwanich**

Department of Computer Engineering, Faculty of Engineering, Chulalongkorn University

pacharawinboon@gmail.com pacharaponarp@gmail.com
arm.supakorn@gmail.com wipu9402@gmail.com ekapolc@cp.eng.chula.ac.th

Abstract

Automatic short answer scoring is a task whose aim is to help grade written works by learners of some subject matter. In niche subject domains with small examples, existing methods primarily utilized similarity-based scoring, relying on predefined reference answers to grade each student's answer based on the similarity to the reference. However, these reference answers are often generated from a randomly selected set of graded student answer, which may fail to represent the full range of scoring variations. We propose a semi-automatic scoring framework that enhances the selective sampling strategy for defining the reference answers through a K-center-based and a K-means-based sampling method. Our results demonstrate that our framework outperforms previous similarity-based scoring methods on a dataset with Thai and English. Moreover, it achieves competitive performance compared to human reference performance and LLMs.

1 Introduction

Automatic short answer scoring is a task that focuses on the development of a system or model capable of grading students' responses to question prompts in educational settings, such as short answers or other text responses (Burrows et al., 2015). This can help reduce the workload for teachers and teaching assistants, particularly when grading homework in large courses.

Machine learning models can be trained to predict the score of a given answer. Researchers have used SVM (Hou et al., 2010), LSTM (Dasgupta et al., 2018), and BERT (Sung et al., 2019) to create such models. However, these require preexisting training data for each questions, which limits the applicability of such methods. Large Language Models (LLMs) have also been explored to score students answers (Lee and Song, 2024). Since LLMs have been trained on a wide range of domains, they can be potentially useful for evaluating student answers in zero-shot and few-shot settings (Chamieh et al., 2024). However, some university-level homework requires specialized technical knowledge, which may fall into domains for which no dedicated LLM has been trained. Fine-tuning an LLM for specific courses presents further challenges, as universities offer many different subjects, making it a significant workload to prepare the necessary datasets for each course. Additionally, LLMs are limited by high resource demands and the cost of API usage (Shekhar et al., 2024).

Another approach is similarity-based scoring (Horbach and Zesch, 2019), where students' answers are compared with a set of reference answers and given the score of the reference answer most similar to their own. Bexte et al. (2023) explored this idea, sampling answers to be manually graded and use as reference with two methods: random sampling and balanced sampling. While the latter showed better performance, it is not applicable in a real grading scenario, since we cannot predetermine the score of each answer to create a balanced reference set for each class. While this could be simulated by having educators create their own reference answer for each score, it becomes quite challenging in higher educations, where more complex and diverse answers are expected.

In this work, we present a semi-automatic, similarity-based scoring framework that eliminates the need for educators to create a separate reference answer set. Instead, educators grade a subset of student answers selected through K-means-based sampling and K-center-based sampling without prior labeling, and the system uses these graded answers as the reference set. Then, we evaluate our similarity-based scoring framework on real data collected from a university in Thailand, which includes Thai, English, and code-switched answers. Our results

*These authors contributed equally to this work.

Proceedings of the Second Workshop in South East Asian Language Processing, pages 27–41
January 20, 2025.

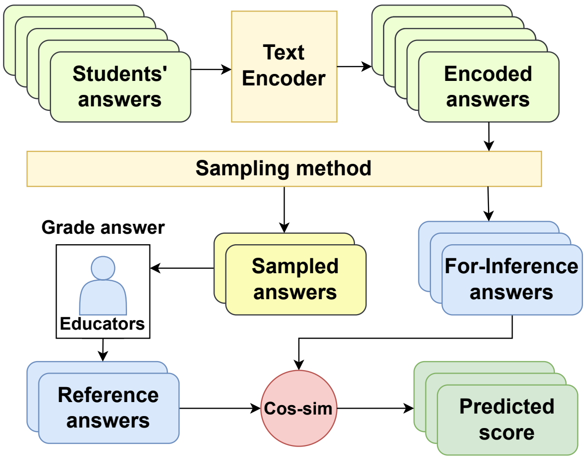

Figure 1: Overview of our semi-automatic, similarity-based scoring framework.

show that this framework outperforms random sampling baseline and achieves performance comparable to human. Our contributions are as follows:

- We propose a semi-automatic, similarity-based scoring framework that uses K-means-based sampling and K-center-based sampling to retrieve diverse reference answers.
- We conduct a comparative study of our similarity-based framework against LLM and human graders by evaluating each method on a bilingual dataset. Besides the typical accuracy-based metrics examined in previous studies, we also proposed the use of consistency-based metrics that measure how consistent a technique would be if performed on the exact same answers.

2 Method

To score a set of student answers, our method consists of two main steps. First, a subset of answers is selected and graded manually to serve as *reference answers*. Then, we assign scores to the rest of the answers by finding the most similar graded answer. An overview of our method is shown in Figure 1.

In order to find the best representative subset of the answers, we can perform some kind of sampling in the text embedding space of the answers. We consider two sampling strategies that aim to maintain the diversity of the sampled subset: a K-means clustering-based strategy and a K-center-based strategy.

2.1 K-means-based Sampling

K-means clustering is a well-known unsupervised method used to classify data by dividing it into a specified number of clusters (MacQueen, 1967), based on Euclidean distance. We utilize this technique to select K representative data points for our reference set. Specifically, for each cluster, we choose the data point closest to the centroid to serve as the reference data.

2.2 K-center-based Sampling

With K-means, the level of similarity in each cluster might vary due to the nature of its algorithm. To ensure that all data point maintain comparably high level of similarity with at least one of the selected reference answer, we instead minimize the maximum distance between any data point and its closest reference answer. This is equivalent to the K-center problem (Hochbaum and Shmoys, 1985), which can be described with the following mixed integer program (MIP).

$$
\begin{aligned}
\min_{x_i, y_{ij}, r} \quad & r \\
\text{s.t.} \quad & \sum_i x_i = K, \qquad x_i \geq y_{ij} \qquad \forall i \forall j \\
& \sum_i y_{ij} \geq 1 \quad \forall j, \quad r \geq d_{ij} y_{ij} \quad \forall i \forall j
\end{aligned} \tag{1}
$$

where x_i is 1 if data point i is used as reference and 0 otherwise, y_{ij} is 1 if the closest reference point from data point j is i and 0 otherwise, r is the maximum cosine distance between any of the points and its closest reference, K is the desired number of reference points, and d_{ij} is the cosine distance between point i and point j. The MIP from eq.1 is computationally prohibitive and various alternatives have been explored (Rana and Garg, 2011). We use an algorithm based on binary search in our experiment, detailed in Appendix D.

After the reference answers are graded, the rest of the answers are scored by selecting the most similar graded answer in the embedding space using cosine similarity.

Course	Prompt	# Answers/prompt
Statistics	Q 1-4	113
Computer	Q 1-2	142
Architecture	Q 3-5	143

Table 1: Number of answers in the dataset.

3 Experimental setup

3.1 Dataset and Human Baseline

We created the dataset by collecting assignment answers from a Computer Architecture course and

Method	% Ref	QWK ↑	MAE ↓	Consistency$_{acc}$ ↑	Consistency$_{err}$ ↓
Human baseline	100%	0.719	0.692	0.620	0.692
Our Similarity-based framework					
Random sampling (Baseline)	30%	0.605	0.708	0.627	0.774
K-means sampling*	30%	<u>0.677</u>	<u>0.639</u>	0.733	0.513
K-center sampling*	30%	0.676	0.656	**<u>0.912</u>**	**<u>0.182</u>**
LLM zero-shot					
Qwen2.5-7B-Instuct	0%	0.356	1.284	0.625	0.288
GPT-4o mini	0%	<u>0.483</u>	<u>1.152</u>	<u>0.749</u>	<u>0.211</u>
LLM few-shot					
Qwen2.5-7B-Instuct	5%	0.443	1.087	0.732	0.215
GPT-4o mini	5%	0.601	0.854	0.719	0.276
Qwen2.5-7B-Instuct	30%	0.597	0.807	0.778	0.217
GPT-4o mini*	30%	**<u>0.691</u>**	**<u>0.619</u>**	<u>0.843</u>	<u>0.198</u>

Table 2: Comparisons of human baseline, similarity-based methods, and LLM approaches. An asterisk (*) indicates that the MAE of that method is significantly better than random sampling using paired t-test ($p < 0.05$). The best results overall are bolded, and the best in each section are underlined.

a Statistics course at a university in Thailand. The dataset contains student responses to nine prompts and their respective official scores, graded by a teaching assistant who was well-acquainted with the topics while following written grading criteria. For any prompt, the students can answer in Thai, English, or a mixture of both. Scores range from 0 to 5, and may include decimal values. These official scores will be used as ground-truth throughout this experiment. Table 1 provides an overview of the number of answers per prompt. The average answer lengths for Statistics and Computer Architecture are 67.79 and 55.92 words, respectively.

Additionally, to simulate the scoring discrepancies that can occurs in a real grading scenario, we had another teaching assistant with similar qualifications grade the responses based on the same criteria. We then compare it with the official score to use as the human baseline for our experiment.

3.2 Evaluation metrics

The main metrics in our experiment are Quadratic Weighted Kappa (QWK) (Cohen, 1968) and Mean Absolute Error (MAE) (Willmott and Matsuura, 2005), which we use to assess the correlation and error between the predicted scores and the ground truth. Note that both metrics are computed on the entire set of answers including the reference answers selected.

All data sampling techniques can give different or multiple possible outcomes. For evaluation, we report the average across different 10 runs.

We also evaluated the consistency of each method by comparing predictions from different runs[1]. **Consistency**$_{acc}$ measures the accuracy between predictions. Two predictions are considered consistent if their absolute difference is under 0.25 (5%). **Consistency**$_{err}$ is equal to the mean absolute error (MAE) between the two predictions.

In addition, to show that our sampling strategy leads to a more diverse representative subset of data, we define a metric called **Representative Score Coverage (RSC)** which is equal to the number of unique scores among the representative samples divided by the total number of unique scores in the dataset. We measured and compared the RSC of each sampling method.

3.3 Experimental Design

We evaluated our framework using three sampling methods: (1) K-means-based sampling, (2) K-center-based sampling and (3) random sampling (baseline), on data encoded using different encoders: (1) Multilingual Universal Sentence Encoder (MUSE) (Yang et al., 2020), (2) gte-Qwen2-7B-instruct (Li et al., 2023), and (3) BGE-M3 (Chen et al., 2024). To simulate workload reduction, we sampled 30% of the data to serve as reference answers and evaluated the performance of each sampling method-encoder combination.

We also assessed the performance of our method in comparison to prompting two LLMs: Qwen2.5-7B-Instruct (Qwen Team, 2024) and GPT-4o mini[2],

[1] consistency metrics for the human baseline is measured using the difference between the two human graders.

[2] gpt-4o-mini-2024-07-18

in both zero-shot and few-shot settings. In the few-shot setup, we randomly selected 5% and 30% of the data as example answers within the prompt.

Furthermore, we also conducted a study to determine the percentage of reference data needed for our framework to surpass the human baseline for each sampling method.

4 Result and Analysis

4.1 Main Results

Table 2 presents the experimental results, with similarity-based methods performance shown being measured on data encoded with MUSE. Both K-means and K-center sampling outperform the random sampling baseline and are comparable to human, showing better performance in MAE but slightly worse in QWK. In the LLM few-shot approach, both LLMs show poor performance for lower number of shots (5%), which is in line with the result presented by Chamieh et al. (2024). After increasing the amount of reference answers to 30% of the data, GPT 4o-mini achieves a performance on par with both our framework and the human baseline. However, our K-center approach shows the best consistency scores overall which is more preferable from a reliability standpoint. We also calculate the RSC for three sampling methods encoded with MUSE. Random sampling achieves an RSC of 0.784, while K-center-based and K-means-based sampling show higher diversity with RSCs of 0.861 and 0.867, respectively.

Method	MUSE	gte-Qwen2	BGE-M3
Random	**31.9%**	35.4%	35.4%
K-means	**27.0%**	30.0%	30.3%
K-center	**25.7%**	32.1%	32.6%

Table 3: Percentage of reference answer needed to achieve MAE lower than human baseline.

Method	MUSE	gte-Qwen2	BGE-M3
Random	**47.8%**	51.3%	51.3%
K-means	**36.4%**	41.8%	40.7%
K-center	**35.4%**	41.1%	40.7%

Table 4: Percentage of reference answer needed to achieve QWK higher than human baseline.

4.2 Additional Results

We also would like to know how many reference answers are needed in order to reach the human baseline. Tables 3 and 4 illustrate the results, showing that the MUSE encoder outperforms the others. On average, K-means sampling achieves the best results in reducing MAE, while K-center sampling performs better in terms of QWK. Figures 2 and 3 show the MAE and QWK scores in relation to the percentage of reference answers for each sampling method, using MUSE as the text encoder.

We also evaluate the performance when the the data is separated by language of answer and by course, the result is presented in Appendix G.

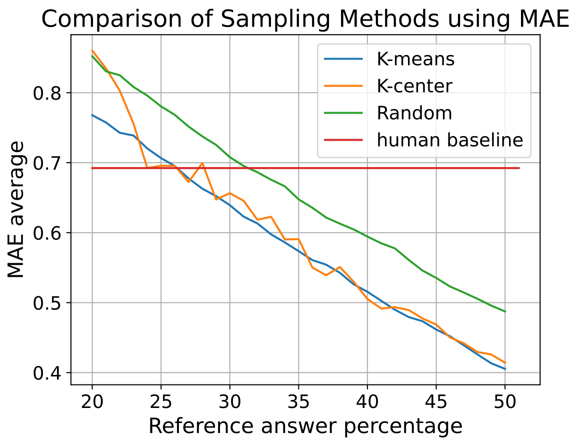

Figure 2: MAE by percentage of reference answers.

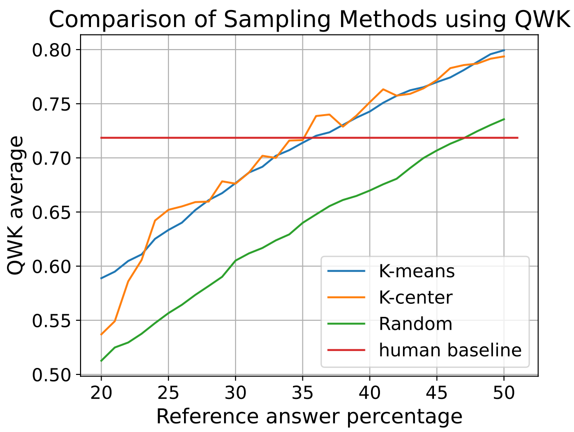

Figure 3: QWK by percentage of reference answers.

5 Conclusion

We propose a semi-automatic, similarity-based scoring framework that employs K-means clustering and K-center sampling to create a reference answer set and conduct a comparative study of our framework against LLM inference and a human baseline. The results demonstrate that our framework outperforms similarity-based scoring methods that use random sampling to create a reference answer set and is comparable to both LLM and human performance.

6 Ethical Considerations

The data contains no personal information, and the graders were compensated fairly for their work.

We would like to note that automatic scoring should be utilized with caution, as it could influence the outcome of the student's grade. Despite the promising MAE, we found that some grading errors could be large. In practice, the automatic grader might be used as a second opinion. The traceable nature of the similarity-based scoring can also be used for spotting errors in human scoring.

7 Limitation

The findings from this study might not be applicable to all subjects and question format. This study is based on two subjects (statistics and computer architecture) which are technical in nature. The answers are around a couple sentences to a paragraph in length. For large language models (LLMs), using a larger set of reference answers might not be feasible with models with limited context. There are certain aspects of this study that might be examined further such as making better use of the reference answers, sampling and grading one answer at a time (active learning), and finetuning the embedding models. MUSE supports Thai, yielding the best results in this study. However, this might not be applicable to other Southeast Asian languages.

Several parts of our framework can be further improved, such as the reference answer selection method, and score assignment. We selected the points closest to the centroids as reference answers based on cosine similarity. However, methods to select the reference answer can also be applied. We also experimented with Euclidean distance which did not significantly affect the results.

References

Marie Bexte, Andrea Horbach, and Torsten Zesch. 2023. Similarity-Based Content Scoring - A more Classroom-Suitable Alternative to Instance-Based Scoring? In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 1892–1903, Toronto, Canada. Association for Computational Linguistics.

Steven Burrows, Iryna Gurevych, and Benno Stein. 2015. The eras and trends of automatic short answer grading. *International journal of artificial intelligence in education*, 25:60–117.

Imran Chamieh, Torsten Zesch, and Klaus Giebermann. 2024. LLMs in Short Answer Scoring: Limitations and Promise of Zero-Shot and Few-Shot Approaches. In *Proceedings of the 19th Workshop on Innovative Use of NLP for Building Educational Applications (BEA 2024)*, pages 309–315, Mexico City, Mexico. Association for Computational Linguistics.

Jianlyu Chen, Shitao Xiao, Peitian Zhang, Kun Luo, Defu Lian, and Zheng Liu. 2024. M3-Embedding: Multi-Linguality, Multi-Functionality, Multi-Granularity Text Embeddings Through Self-Knowledge Distillation. In *Findings of the Association for Computational Linguistics: ACL 2024*, pages 2318–2335, Bangkok, Thailand. Association for Computational Linguistics.

Jacob Cohen. 1968. Weighted kappa: Nominal scale agreement provision for scaled disagreement or partial credit. *Psychological bulletin*, 70(4):213.

Tirthankar Dasgupta, Abir Naskar, Lipika Dey, and Rupsa Saha. 2018. Augmenting Textual Qualitative Features in Deep Convolution Recurrent Neural Network for Automatic Essay Scoring. In *Proceedings of the 5th Workshop on Natural Language Processing Techniques for Educational Applications*, pages 93–102, Melbourne, Australia. Association for Computational Linguistics.

Dorit S Hochbaum and David B Shmoys. 1985. A best possible heuristic for the k-center problem. *Mathematics of operations research*, 10(2):180–184.

Andrea Horbach and Torsten Zesch. 2019. The influence of variance in learner answers on automatic content scoring. In *Frontiers in education*, volume 4, page 28. Frontiers Media SA.

Wen-Juan Hou, Jia-Hao Tsao, Sheng-Yang Li, and Li Chen. 2010. Automatic assessment of students' free-text answers with support vector machines. In *Trends in Applied Intelligent Systems: 23rd International Conference on Industrial Engineering and Other Applications of Applied Intelligent Systems, IEA/AIE 2010, Cordoba, Spain, June 1-4, 2010, Proceedings, Part I 23*, pages 235–243. Springer.

Jung X Lee and Yeong-Tae Song. 2024. College Exam Grader using LLM AI models. In *2024 IEEE/ACIS 27th International Conference on Software Engineering, Artificial Intelligence, Networking and Parallel/Distributed Computing (SNPD)*, pages 282–289. IEEE.

Zehan Li, Xin Zhang, Yanzhao Zhang, Dingkun Long, Pengjun Xie, and Meishan Zhang. 2023. Towards general text embeddings with multi-stage contrastive learning. *arXiv preprint arXiv:2308.03281*.

J MacQueen. 1967. Some methods for classification and analysis of multivariate observations. In *Proceedings of 5-th Berkeley Symposium on Mathematical Statistics and Probability/University of California Press*.

Qwen Team. 2024. Qwen2.5: A Party of Foundation Models.

Rattan Rana and Deepak Garg. 2011. An evaluation of k-center problem solving techniques, towards optimality. *International Journal of Advancements in Technology*, 2:206–214.

Shivanshu Shekhar, Tanishq Dubey, Koyel Mukherjee, Apoorv Saxena, Atharv Tyagi, and Nishanth Kotla. 2024. Towards Optimizing the Costs of LLM Usage. *arXiv preprint arXiv:2402.01742*.

Chul Sung, Tejas Dhamecha, Swarnadeep Saha, Tengfei Ma, Vinay Reddy, and Rishi Arora. 2019. Pre-Training BERT on Domain Resources for Short Answer Grading. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 6071–6075, Hong Kong, China. Association for Computational Linguistics.

Cort J Willmott and Kenji Matsuura. 2005. Advantages of the mean absolute error (MAE) over the root mean square error (RMSE) in assessing average model performance. *Climate research*, 30(1):79–82.

Yinfei Yang, Daniel Cer, Amin Ahmad, Mandy Guo, Jax Law, Noah Constant, Gustavo Hernandez Abrego, Steve Yuan, Chris Tar, Yun-hsuan Sung, Brian Strope, and Ray Kurzweil. 2020. Multilingual Universal Sentence Encoder for Semantic Retrieval. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, pages 87–94, Online. Association for Computational Linguistics.

A Visualization of the grading Framework

Figure 4 illustrates how our framework selects reference answers and assigns scores to other answers. After encoding all answers into dense vectors, sampling strategies were employed to select a diverse subset of answers for human grading. Subsequently, all other answers were assigned the same score as their most similar reference answer.

B Additional Dataset Information

The data was taken from homework assignments in two courses namely, Statistics and Computer Architecture. Both courses were held at a university in Thailand during 2023. Students completed the assignments by filling out the provided text boxes in the university's learning management platform. All answers were marked by hand in accordance with predetermined rubrics.

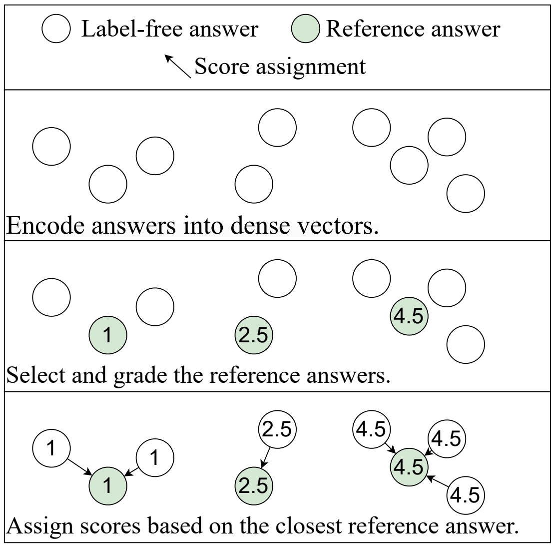

Figure 4: Visualization of how our similarity-based framework operates.

C Question Information

C.1 Statistics Course

These questions covers the topic of Statistics and A/B Testing. In this question set, a situation is described, followed by 4 questions which are based on it. The questions are given in Thai, but students are allowed to answer in either English or Thai. The situation and one example question is shown in Table 5, along with translation. Table 6 shows the corresponding rubric.

The rubric for each question is defined based on the topics which a full-score answer should cover. And for each topic the answer covers, a partial score will be given if the answer expresses that topic correctly in accordance with the rubric. The partial scores in each rubric are then summed into the final score. Figure 6 shows the score distribution for each question.

To demonstrate how the answers are marked, Tables 7 and 8 show answers from 2 students with translations, along with how the answers perform in each rubric, and the score received.

C.2 Computer Architecture Course

These questions cover the general knowledge about computer architecture and the changes in computer architecture throughout the ages.

In this homework, students are required to read a short article and answer questions regarding the article, mainly asking for explanations to certain topics. The article is "A New Golden Age for Computer Architecture" by John L. Hennessy and David

A. Patterson. One of the questions is shown in Table 9 as an example.

The answers to all questions can be found in the article, and we expect the students to read it in order to be able to answer the questions. Therefore, a good answer in this question set should address all the sub-questions along with sufficient supporting evidence from the article. The questions are designed to be self-contained within the article, and no extra scores are given should the student include information from other sources.

To grade the question in Table 9, the rubrics in Table 10 are used. Table 11 and 12 show examples of students' answers and example grading logic. The score distribution for each question is shown in Figure 7.

D Algorithm for Solving K-center

We can use binary search to find the optimal r by testing the feasibility of the following integer program.

$$\begin{aligned} \text{Feasible}(d_{ij}, K, r): & \sum_i x_i = K, \\ & \sum_{l \in C_i} x_l \geq 1, \forall i \\ & C_i = \{j \mid r > d_{ij}\} \end{aligned} \tag{2}$$

where x_i is 1 if data point i is used as reference and 0 otherwise, k is the desired number of reference points, d_{ij} is the cosine distance between point i and point j, and r is the maximum cosine distance allowed between any of the points and its closest reference.

Since the infeasibility of this integer program implies that r is too small for the given K, we can use binary search to iteratively find the minimum r.

The resulting r can be used to determine the optimal reference points. If there are multiple possible solutions, we randomly select one. We denote this technique, mixed integer linear program with binary search K-center algorithm (MBK-Center) which is detailed in Algorithm 1.

E LLMs inference

Figures 8 – 9 present the prompt templates used for the inference of GPT-4o-mini and Qwen2.5-7B-Instruct in both zero-shot and few-shot settings, correspondingly.

Algorithm 1 Mixed integer linear program with Binary search K-center (MBK-Center)

```
ub ← 2            ▷ initial upper bound of Cos-Dist (ub)
lb ← 0            ▷ initial lower bound of Cos-Dist (lb)
while ub ≠ lb do
    r ← (ub+lb)/2
    if Feasible(d_ij,K,r) then        ▷ From Eq. 2
        ub ← r
    else
        lb ← r
    end if
end while
return r
```

F Cluster Homogeneity Analysis

Figure 5 shows example distributions of the actual scores of answers assigned to different reference solutions in the clustering process. Most groups contain similar scores. The differences to the reference answer scores are typically less than one. This supports the validity of similarity-based scoring. However, some groups exhibit high variance in true scores. In many cases, these discrepancies are due to: 1) the answer being difficult to grade, resulting in significantly different scores even when graded by humans, 2) grading errors leading to incorrect true scores. We believe that identifying and addressing such cases will be crucial in improving automatic answer scoring systems.

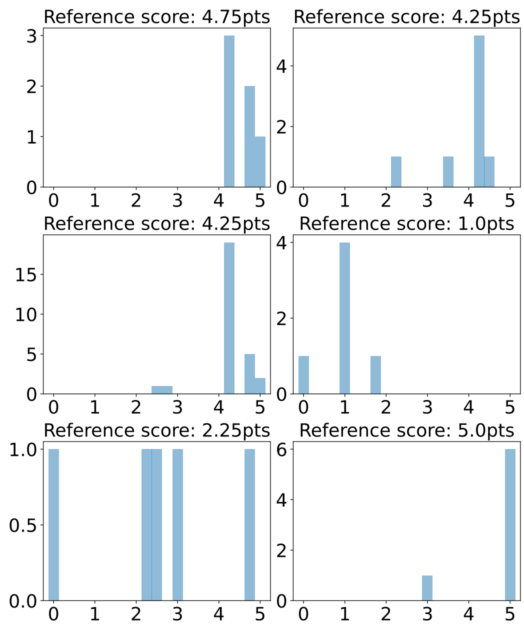

Figure 5: Example of a histogram showing frequency of scores in each cluster using K-means-based sampling on MUSE-encoded data.

G Additional Experimental Results

Tables 13 – 15 present a performance comparison between different input settings, using different sampling methods (K-means and K-center), with data encoded using MUSE, gte-Qwen2-7B-instruct, and BGE-M3. The QWK and MAE are measured when using data from each course in two settings: (1) inputting all answers, (2) inputting only answers in a single language. The percentage of reference answers used is 30%. Note that the performance on English answers for the Statistics course is not measured due to the low number of answers.

	Language	Situation
Situation	TH	Hamtube เป็นแพลตฟอร์มดูวีดีโอออนไลน์ ที่อนุญาตให้ผู้ใช้อัปโหลด แชร์ และดูวิดีโอได้ แฮมทาโร่ เป็นหัวหน้าทีมการตลาดของ Hamtube และเขาต้องการทราบว่าการย้ายตำแหน่งของโฆษณาจะช่วยเพิ่มยอดขาย (ผู้ใช้คลิกโฆษณามากขึ้น) หรือไม่ ดังนั้นเขาตัดสินใจที่จะดำเนินการทดลอง A/B testing.
	EN	Hamtube is an online video platform on which users can watch, share, and upload videos. Hamtaro, the head of marketing for Hamtube, is eager to know whether the new advertisement position would increase sales (or would increase the clickthrough rate). Thus, Hamtaro decided to conduct an A/B test to prove this statement.
Question	TH	แฮมทาโร่จะต้องเลือกว่าอยากให้สัดส่วนของ user ที่เห็นโฆษณาตำแหน่งเก่า ต่อ user ที่เห็นโฆษณาตำแหน่งใหม่เป็นเท่าไร โดยตอนนี้แฮมทาโร่กำลังลังเลระหว่างสัดส่วน 50/50 กับ สัดส่วน 80/20 จงอธิบายข้อดีข้อเสียของการเลือกสัดส่วนแต่ละแบบ และตอบว่าแบบใดที่น่าจะเหมาะสมกับปัญหานี้มากกว่า
	EN	Hamtaro is deciding the ratio between users who would see the current ad position, and the newly proposed one. He is considering a 50/50 ratio, or an 80/20 ratio. Explain the pros and cons of each decision and choose the ratio which is more suitable for this problem.

Table 5: Situation and example question from the Statistics course with translation.

No	Language	Rubric	Score by Rubric	Full Score
1	TH	อธิบายเกี่ยวกับระยะเวลาทดลองว่าถ้าแบ่งกลุ่มแบบ 50/50 จะทำให้เราได้ผลการทดลองเร็วขึ้น เทียบกับ 80/20	2	5
	EN	Explain about the speed of the experiment, where the 50/50 ratio would yield results faster, and the 80/20 ratio would result in a longer experiment.		
2	TH	อธิบายเกี่ยวกับความเสี่ยงต่อธุรกิจจากการทดลอง ก็คือการแบ่ง 50/50 จะมีความเสี่ยงต่อธุรกิจมากกว่า (เช่นส่งผลให้ยอดขายอาจลดลงมากกว่า) ส่วน 80/20 จะมีความเสี่ยงน้อยกว่า	2	
	EN	Explain about the risk associated with the experiment, where the 50/50 ratio could provide higher risks (such as lower sales) while the 80/20 ratio results in lower risk.		
3	TH	ตอบว่าสัดส่วนแบบไหนดีกว่า โดยอ้างอิงเหตุผลจากที่ตอบมาก่อนหน้า (สามารถตอบ 50/50 หรือ 80/20 ได้ทั้งคู่ แต่หากตอบแบบครึ่ง ๆ กลาง ๆ จะได้ 0)	1	
	EN	Answer which ratio would be better with reasonable arguments. (Either 50/50 or 80/20 is fine. However, indecisive answers would get 0 points)		

Table 6: Rubric for the example question in Table 5 with translation.

Original Answer in Thai	การใช้สัดส่วน 50/50 นั้น จะใช้เวลาทดสอบน้อยกว่า 80/20 เนื่องจากมีการ split จำนวนให้ทั้ง 2 versions เยอะ ทำให้จำนวน user (ของทั้ง 2 versions) ถึงยอดที่ต้องการโดยเร็ว ในทางกลับกัน หาก version ใหม่ที่ทดสอบ มีสิ่งที่แตกต่างจาก version default เยอะ หาก version ใหม่ไม่เวิร์ค user ก็จะได้รับผลกระทบด้านลบมากขึ้นตาม ดังนั้นการแบ่ง 80/20 ก็จะดีกว่าในแง่ของการลดความเสี่ยง ทั้งนี้สำหรับ Hamtube การย้ายตำแหน่งของโฆษณาเพียงอย่างเดียวนั้นอาจไม่ได้ส่งผลกระทบด้านลบที่ใหญ่หลวงมากมาย (หากไม่ได้ทำอะไรสุดโต่ง) ดังนั้นการเลือกแบ่ง 50/50 จึงเหมาะสมกว่าเนื่องจากใช้เวลาน้อยกว่า และ ความเสี่ยงที่อาจเกิดขึ้นสามารถรับได้	
Translated Answer	Using a 50/50 split will require less testing time than an 80/20 split because it allows for a larger number of users to experience both versions, reaching the desired user count more quickly. On the other hand, if the new version being tested has significant differences from the default version and new version doesn't work, it could have a greater negative impact on users. Therefore, an 80/20 split is better in terms of risk reduction. However, for Hamtube, simply moving the ad placement may not lead to significant negative impacts (as long as it's not an extreme change). Thus, choosing a 50/50 split is more suitable due to the shorter testing time and manageable risk.	
Rubric No	Reason	Score
1	The answer mentioned that a 50/50 split would require less testing time since it would make the treatment group reach its user count goals faster. Thus, this answer gets 2 points in this criterion.	2
2	The answer mentioned that while the 50/50 group took less experiment time, if the new version launched has a negative impact, it would impact more users. This makes the 80/20 group a safer choice. Thus, this answer gets 2 points in this criterion.	2
3	The student decided that the risks for this experiment were not high and still manageable. Therefore, the merits of a faster experiment outweighed the risks, and the student chose the 50/50 group. Since this answer decisively chose the 50/50 group with reasonable supporting arguments, it gets 1 point in this criterion.	1
	Full Score	5

Table 7: First example answer for the question in Table 5 with its grading comments and translation.

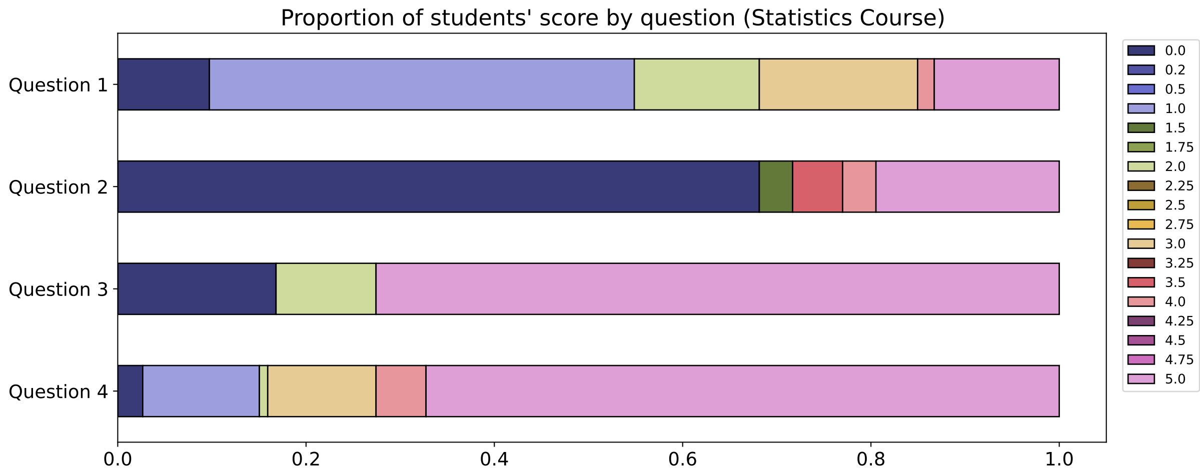

Figure 6: Proportion of students' score by question in Statistics course.

Original Answer in Thai	ถ้าเลือกแบบ 50/50 จะเสี่ยงกว่าเพราะถ้าทำให้ user 50% ที่เจอโฆษณาที่ตำแหน่งใหม่ตัดสินใจคลิ้กน้อยลงจะทำให้ยอดตกลงมากกว่า จึงควรเลือก 80/20 เพื่อเป็นการลดความเสี่ยงจนเรามั่นใจว่ายอดเพิ่มจริงๆถึงขยับเปอเซนต์ขึ้น	
Translated Answer	Choosing the 50/50 option is riskier because if 50% of users who see the ad in the new position decide to click less, the revenue could drop significantly. It’s better to go with an 80/20 split to reduce the risk until we’re confident that the revenue is genuinely increasing before adjusting the percentage further.	
Rubric No	Reason	Score
1	The question does not mention anything about the testing time. Thus, this answer gets no points in this criterion.	0
2	The answer mentioned that the 50/50 group might cause revenue to plummet (since more users saw the hypothetically worse treatment group). This makes the 80/20 group a safer choice. Thus, this answer gets 2 points in this criterion.	2
3	The student chose the 80/20 due to it being a safer choice. Although he did not consider the shorter testing time by the 50/50 group. This makes a reasonable conclusion based on the student’s observation. Thus, this answer gets 1 point in this criterion.	1
	Full Score	3

Table 8: Second example answer for the question in Table 5 with its grading comments and translation.

	Content
Question	Explain why DSAs can achieve higher performance and greater energy efficiency.

Table 9: Example question from the Computer Architecture course.

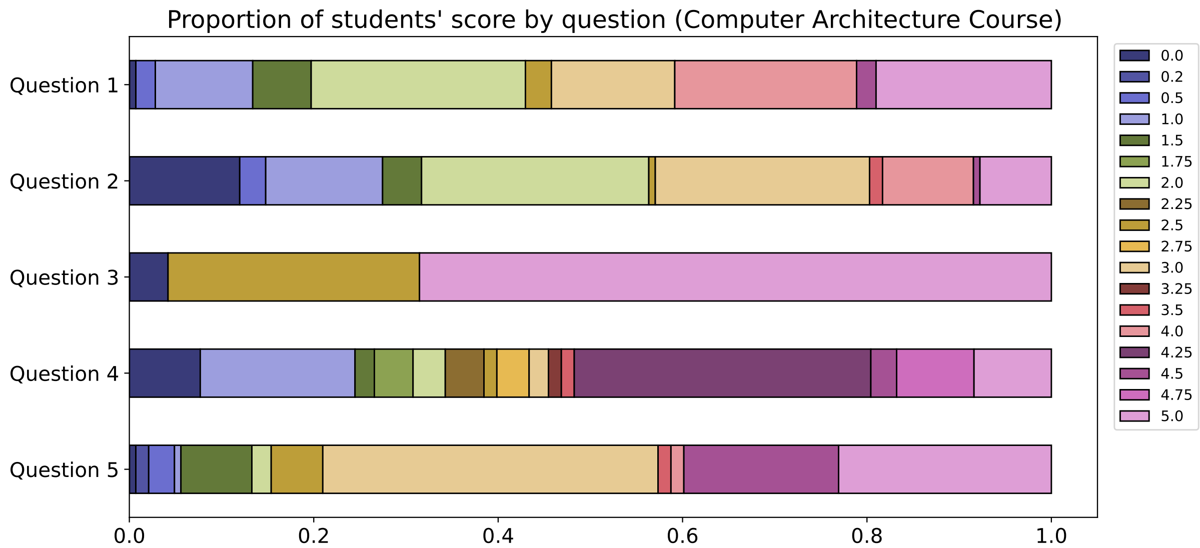

Figure 7: Proportion of students’ score by question in Computer Architecture Course.

No	Rubric	Score by Rubric	Full Score
1	Discuss specialization on one of the four following topics. A bonus point if at least one of the topics in the rubric below is correctly explained.	1	5
2	Parallelism: Explain about DSAs using the most effective form of parallelism for that domain. While also giving an example e.g., • SIMD is faster than MIMD but less flexible • VLIW is better for explicitly parallel programs 0.25 points given are for the example. i.e., if the answer explains this rubric, but no examples are given, it shall get 0.75 points.	1	
3	Memory hierarchy is given 1 point is given if at least one of the following is discussed • memory access uses much more energy than computation • cache doesn't work well when the datasets are large • cache works well when the locality is high • in applications where the memory access patterns are well defined and discoverable at compile time, user-controlled memories use less energy than cache 0.5 points is given if memory hierarchy is mentioned but the stated concepts are not dicussed.	1	
4	Explain that DSAs can use less precision for some specific works (e.g., machine learning).	1	
5	Explain that DSAs benefit from targeting programs written in domain-specific languages.	1	

Table 10: Rubric for the example question in Table 9.

Example Answer	DSA or Domain-specific architecture can achieve better performance because they are more closely tailored to the needs of the application. There are 4 main reasons behind these, 1. DSAs exploit a more efficient for of parallelism for the specific domain 2. DSAs can make more effective use of the memory hierarchy. 3. DSAs can use less precision when it is adequate 4. DSAs benefit from targeting programs written in domain-specific languages (DSLs) that expose more parallelism	
Rubric No	Reason	Score
1	One of the reasons below is valid. Thus, it receives 1 point from this criterion.	1
2	The answer mentions parallelism but did not give an example. Thus, it receives 0.75 points from this criterion.	0.75
3	The answer mentions the more effective use of the memory hierarchy but does not provide any more details. Thus, it receives 0.5 points from this criterion.	0.5
4	The answer explains that DSAs can use less precision. Thus, it receives 1 point from this criterion.	1
5	The answer explains that DSAs benefit from targeting programs. Thus, it receives 1 point from this criterion.	1
	Full Score	4.25

Table 11: First example answer for the question in Table 9 with its grading comments.

Example Answer	DSAs can achieve higher performance form of parallelism for the specific domain. Typically, DSAs use SIMD which is more efficient than MIMD because it needs to fetch only one instruction stream, and processing units operate in lockstep. DSAs can achieve greater energy efficiency because of the effective use of the memory hierarchy. Due to the memory access patterns being well-defined and discoverable at compile time, programmers and compilers can optimize the use of the memory better than dynamically allocated caches.	
Rubric No	Reason	Score
1	One of the reasons below is valid. Thus, it receives 1 point from this criterion.	1
2	The answer mentions parallelism, and also stated that DSAs use SIMD which is more efficient than MIMD as an example. Thus, it receives 1 point from this criterion.	1
3	The answer mentions the more effective use of the memory hierarchy due to the memory access patterns being well-defined. Thus, it receives 1 point from this criterion.	1
4	The answer does not cover the fact that DSAs can use less precision. Thus, it receives no points from this criterion.	0
5	The answer does not cover the fact that DSAs benefit from targeting programs written in domain-specific languages. Thus, it receives no points from this criterion.	0
	Full Score	3

Table 12: Second example answer for the question in Table 9 with its grading comments.

Grade the student's answer based on the criteria, and return a final score as a single number between 0 and {max_score}**. Make sure to provide only the numerical score without any additional explanation.**

Question:
{question}

Criteria:
{criteria}

Max score:
{max_score}

Student answer:
{answer}

Final score:

Figure 8: Zero-Shot grading prompt template.

Grade the student's answer based on the criteria, and return a final score as a single number between 0 and {max_score}**. Make sure to provide only the numerical score without any additional explanation.**

Question:
{question}

Criteria:
{criteria}

Max score:
{max_score}

Example answer:
Student answer: {ref_answer_1}
Final score: {label_1}
…
Student answer: {ref_answer_n}
Final score: {label_n}

Student answer:
{answer}

Final score:

Figure 9: Few-Shot grading prompt template.

Course/Method	QWK		MAE	
	K-means	**K-center**	**K-means**	**K-center**
Statistics				
All answers	0.641	0.587	0.722	0.830
Thai Answers	0.634	0.616	0.731	0.792
Computer Architecture				
All answers	0.706	0.748	0.573	0.518
English Answers	0.724	0.749	0.541	0.525
Thai answers	0.354	0.350	0.866	0.843

Table 13: Performance comparison between different input settings, on MUSE-encoded data.

Course/Method	QWK		MAE	
	K-means	**K-center**	**K-means**	**K-center**
Statistics				
All answers	0.613	0.553	0.728	0.808
Thai Answers	0.604	0.558	0.735	0.830
Computer Architecture				
All answers	0.644	0.661	0.647	0.644
English Answers	0.653	0.642	0.638	0.687
Thai answers	0.455	0.408	0.707	0.803

Table 14: Performance comparison between different input settings, on gte-Qwen2-7B-instruct-encoded data.

Course/Method	QWK		MAE	
	K-means	**K-center**	**K-means**	**K-center**
Statistics				
All answers	0.570	0.529	0.816	0.876
Thai Answers	0.562	0.535	0.826	0.870
Computer Architecture				
All answers	0.703	0.682	0.583	0.634
English Answers	0.723	0.677	0.558	0.666
Thai answers	0.472	0.480	0.735	0.762

Table 15: Performance comparison between different input settings, on BGE-M3-encoded data.